Ultra-Fast Deterministic Approximate Near Neighbor Search in High Dimensions

Abstract

We study the $(1+\eta, r)$-approximate near neighbor problem (ANN) in high-dimensional Euclidean space, focusing on the "ultra-fast" regime: data structures of size polynomial in n and d that answer queries in $d \cdot \operatorname{poly}\log(n, 1/\eta)$ time. Two classic results anchor this regime. The [IM98] data structure achieves linear-in-d query time but is only correct per query with high probability over its construction, so its failure probability cannot be amplified without rebuilding. The [KOR98] data structure allows amplification via query randomness but pays a quadratic dependence on d. We resolve the natural question left open by these works, giving a data structure of $\operatorname{poly}(n, d)$ space whose query algorithm is deterministic and runs in worst-case time $d \cdot \operatorname{poly}\log(n, 1/\eta)$, correct simultaneously for all queries.

1 Introduction

In the *near neighbor search* problem, we are given a set X of n points in d-dimensional Euclidean space, and the goal is to build a data structure that, given a query point q, reports a point of X within distance r of q under the Euclidean metric, if one exists. This problem arises throughout computer science, with applications in databases, data mining, information retrieval, computer vision, and signal processing. Efficient algorithms are known when the dimension d is "low" (e.g., [Mei93], building on [Cla88]). However, despite decades of effort, current solutions suffer from the *curse of dimensionality*, i.e., space or query time exponential in d . To overcome this, starting in the late 1990s, researchers proposed *approximation* algorithms [KOR98, IM98]. In the $(1+\eta, r)$-approximate near neighbor problem (ANN), the data structure may return any point whose Euclidean distance from the query is at most $(1+\eta)r$, for an approximation factor $1+\eta > 1$. Many such algorithms are known, offering tradeoffs among approximation factor, space, and query time.

In this paper we focus on the "ultra-fast" endpoint of this tradeoff, namely data structures with very fast query time, while preserving space polynomial in n and d (for any constant η). In fact, this was the focus of the earliest papers in the late-1990s wave of results [KOR98, IM98]. Both papers give algorithms with $n^{\mathrm{poly}(1/\eta)}$ space and the following probabilistic query-time guarantees:

- **[IM98]:** A $d \cdot \mathrm{polylog}(n, 1/\eta)$ query-time bound, with the following correctness guarantee: for each query point q, the data structure is correct with probability $1 - 1/\mathrm{poly}(n)$, taken over the randomness generated during the *construction* of the data structure.
- **[KOR98]:** A $d^2 \cdot \mathrm{polylog}(n, 1/\eta)$ query-time bound, with the following correctness guarantee: for each query point q, the data structure is correct with probability $1 - 1/\mathrm{poly}(n)$, taken over the randomness generated during the *query* procedure.

This presents a natural tradeoff. On the one hand, the guarantee of [KOR98] is stronger, as the failure probability can be reduced by repeating the query procedure; this cannot be done for [IM98], since doing so would require rebuilding the data structure. On the other hand, the query-time bound of [IM98] has only linear dependence on d.

This leads to a natural question: is there a $\mathrm{poly}(n, d)$-size data structure that guarantees $d \cdot \mathrm{polylog}(n, 1/\eta)$ query time in the worst case, for all queries? Perhaps surprisingly, to the best of our knowledge there has been no further progress on this question since the original work of [KOR98] and [IM98].

1.1 Our results

In this work, we resolve this natural question, giving a data structure of $\mathrm{poly}(n, d)$ space whose query algorithm is deterministic and runs in worst-case time $d \cdot \mathrm{poly}\log(n, 1/\eta)$, correct simultaneously for all queries. Formally, our main result is the following.

Theorem 1.1. *Let $n, d \in \mathbb{Z}_{>0}$ and let $\eta \in (0, 1/2)$, such that $n \gg d \gg \mathrm{poly}(\log n, \eta^{-1})$. There exists a randomized algorithm $\mathcal{A}$ such that given a pointset $X \subset \mathbb{R}^d$ of size n, with*

0.9 *probability, algorithm* $\mathcal{A}$ *outputs a data structure with a function* $\textsc{Query} : \mathbb{R}^d \to X$ *that satisfies:*

$$\forall q \in \mathbb{R}^d : \|q - \textsc{Query}(q)\|_2 \le (1 + \Theta(\eta)) \min_{x \in X} \|q - x\|_2.$$

The data structure takes $\mathrm{poly}(n, d)$ *space for a fixed* η*. The* Query *function is deterministic and runs in time*

$$d\,\mathrm{poly}(\log n, \eta^{-1}).$$

Crucially, our data structure uses randomness only for the initialization and preprocessing the database. Once the data structure is successfully initialized and preprocessed, it correctly responds to *every possible query* $q \in \mathbb{R}^d$ deterministically in linear time in d.

1.2 Our techniques

Our data structure is composed of $t = \Theta(d \log n)$ independent copies of a randomized ANN data structure in the lower dimension $\mathrm{poly}(\log n, \eta^{-1})$. To construct these low-dimensional ANN data structures, we combine the randomized mapping of [FI26] with the ANN data structure of [IM98]. As mentioned in the introduction, [IM98] gives a data structure whose query algorithm is deterministic, but only succeeds with high probability for each fixed query, where the probability is over the randomness of constructing the data structure. We show that, using sufficiently many low-dimensional projections of the inputs via the map of [FI26], this point-wise guarantee can be upgraded to a guarantee that holds simultaneously for all queries.

Outline of the data structure. We start with the randomized linear map M from [FI26]. The output Mx is divided into $K = \mathcal{O}(d)$ consecutive blocks $z_1(x), \cdots, z_K(x)$. The crucial property is that, once M is successfully generated, simultaneously for every $x \in \mathbb{R}^d$, the squared norms of these blocks are close to $\|x\|_2^2$ on average:

$$\sum_{k=1}^{K} \left| \|z_k(x)\|_2^2 - \|x\|_2^2 \right| \le \frac{K}{\mathrm{poly}(\log n, \eta^{-1})} \|x\|_2^2. \tag{1}$$

Moreover, each block has dimension $\mathrm{poly}(\log n, \eta^{-1})$, and Mx can be computed in near-linear time in d. In the rest of this section, we condition on M being successfully generated.

For each construction iteration $i \in [t]$, where $t = \Theta(d \log n)$, we independently sample $\Theta(\log n/\varepsilon^2)$ blocks of Mx and concatenate these sampled blocks. We use the same M and block-sampling for every database points $x \in X$. Finally, we build a low-dimensional ANN data structure of [IM98] on the resulting projected database. Given a query q, we compute Mq, query all t low-dimensional data structures, and obtain candidates $p_1, \cdots, p_t \in X$.

In other words, we use M as a mixing procedure, which produces many low-dimensional projections. For any q and any x, the squared distance $\|q - x\|_2^2$ is roughly preserved by a large fraction of these projections. Thus the hope is that at least some of the data structures built against the projected databases will output a correct candidate.

Point-wise correctness. Fix a query q. We say that one iteration $i \in [t]$ is *good* for q if three events occur. First, the sampled projection does not significantly increase the distance from q to its nearest neighbor in the database. Second, it does not significantly decrease the distance from q to any database point. Third, the ANN data structure of [IM98] is correctly constructed for this query q's corresponding projection.

If an iteration i is good, then the correctness of [IM98] immediately implies that its candidate p_i is an approximate nearest neighbor in the original space. Thus, the main question is how likely an iteration is to be good. Using Eq. (1) we show the point-wise guarantee that for every fixed query, a random iteration is good with probability at least 0.9.

From point-wise to for-all. One difficulty is that the above argument only applies to a fixed q. And it is unclear how to union bound over all $q \in \mathbb{R}^d$, e.g., a standard ε-net discretization argument over the unit sphere does not seem to work, since ε-close queries on the sphere can have very different nearest neighbors.

To overcome this difficulty, we shift the discretization from the query space to the space of randomness. Let Γ denote the space of randomness of sampling the projection and constructing the [IM98] data structure for one iteration. And for every query q, define

$$G_q := \{\gamma \in \Gamma : \gamma \text{ defines a } \textit{good} \text{ iteration for } q\}.$$

This is well-defined since whether an iteration is good is completely determined by the randomness used to initialize this iteration. The point-wise argument above says that the probability measure of every G_q is at least 0.9. We then show that the family of sets

$$\mathcal{G} := \{G_q : q \in \mathbb{R}^d\}$$

has a VC-dimension only $\mathcal{O}(d \log n)$. This follows from the observation that whether each γ is good for a query q is completely determined by the signs of degree-2 polynomials that check whether the (squared) distance preservation properties are met. Counting the number of these polynomials and applying a bound on realizable sign patterns of low-degree polynomials [AS88] then implies the VC-dimension bound.

Therefore, we can apply the uniform convergence theorem for families of bounded VC dimension [VC71, BEHW89]. After taking $t = \Theta(d \log n)$ independent iterations, with high probability the empirical fraction of good iterations is simultaneously close to its expectation for every query $q \in \mathbb{R}^d$. In particular, we obtain the stronger guarantee that with high constant probability over $\gamma_1, \cdots, \gamma_t$,

$$\forall q \in \mathbb{R}^d : \qquad \frac{1}{t}\Big|i \in [t] : \gamma_i \text{ is good for } q\Big| \geq 0.8.$$

Efficient outputting. It remains to efficiently select a good candidate among $p_1, \ldots, p_t$. Computing all true distances $\|q - p_i\|_2$ would take quadratic time in d. Instead, we reuse the sampled blocks above to obtain fast estimates of these distances. Using a similar point-wise argument followed by uniform convergence, we show that, simultaneously for every query q, a 0.95-fraction of the sampled block sets yield median-based estimates that accurately approximate the distance from q to every database point.

We then use these estimates in a halving procedure. In each round, we divide the sampled block sets evenly among the current candidates. Each candidate is scored using only its assigned block sets, and we retain the half of candidates with the smallest scores. Since at least a 0.8-fraction of the initial candidates are good and only a 0.05-fraction of the sampled block sets are inaccurate, a constant fraction of good candidates survives every round, with only a $1 + \mathcal{O}(\eta/\log t)$ loss in the approximation factor. After $\log t$ rounds, one candidate remains, with total approximation loss $1 + \mathcal{O}(\eta)$. The total running time of this procedure is $d \cdot \mathrm{poly}(\log n, \eta^{-1})$.

2 Our Data Structure

Notations. For any integer $n \in \mathbb{Z}_{>0}$, we use $[n]$ to denote the set of integers $\{1, \cdots, n\}$. For any integer $m \in \mathbb{Z}_{>0}$, any vector $x \in \mathbb{R}^m$, and any subset $T \subseteq [m]$, we use $x_T \in \mathbb{R}^{|T|}$ to denote a subvector of x indexed by the set T. Similarly, when T is an ordered multi-set (i.e. a sequence), $x_T \in \mathbb{R}^{|T|}$ denotes a vector consisting of coordinates of x, indexed by elements in T in the specified order.

2.1 Construction with a quadratic output time

In this section, we describe our algorithm, which satisfies the guarantees in Theorem 1.1, except that the last output line (Line 13) naively computes $t = \mathcal{O}(d \log n)$ Euclidean distances in $\mathbb{R}^d$, which takes quadratic time in d. In Section 2.3, we will describe how to replace this naive outputting step with a halving procedure that runs in linear time in d.

Let $n, d \in \mathbb{Z}_{>0}$ and $n \gg d \gg \mathrm{poly}\log n$. Throughout this paper, we assume that the distances between all relevant vectors are unique and non-zero.

Lemma 2.1 ([FI26])**.** *For any C such that $\log d \le C \le d$, there exist integers $K = \mathcal{O}(d)$, $l_1, \cdots, l_K = \mathcal{O}(C^3 \,\mathrm{poly}\log d)$ and a randomized algorithm $\mathcal{A}$, such that with .99 probability, algorithm $\mathcal{A}$ outputs a linear map $\mathcal{M} : \mathbb{R}^d \to \mathbb{R}^{\sum_{k=1}^K l_k}$ with the following guarantee:*

For all $x \in \mathbb{R}^d$, $\mathcal{M}x$ can be written as $\mathcal{M}x = \oplus_{k=1}^K z_k(x)$ for block vectors $z_k(x) \in \mathbb{R}^{l_k}$ depending on x, such that

$$\sum_{k=1}^{K} \left| \|z_k(x)\|_2^2 - \|x\|_2^2 \right| \le \frac{K}{C} \|x\|_2^2.$$

Moreover, $\mathcal{M}x$ can be computed in $\mathcal{O}(Cd \log d + C^3 d \log C)$ time.

In English, $\mathcal{M}x$ can be partitioned into K consecutive blocks, and each block has a squared norm close to the squared norm of x.

In the following, we denote $L_k := \sum_{i=1}^k l_i$ for each $k \in [K]$. In particular, the output dimension of $\mathcal{M}$ becomes $L_K = \sum_{k=1}^K l_k = \mathcal{O}(d \cdot C^3 \,\mathrm{poly}\log d)$. We also denote the index sets

$$I_k := \{i \in \mathbb{Z}_{>0} : L_{k-1} < i \le L_k\}$$

for each $k \in [K]$, where L_0 is interpreted as 0. In this way, for any $x \in \mathbb{R}^d$ and any $k \in [K]$, the block vector $z_k(x)$ can be formally identified as the subvector $(\mathcal{M}x)_{I_k}$.

Theorem 2.2 ([IM98])**.** *Given a dimension parameter $m \in \mathbb{Z}_{>0}$, a database size $n \in \mathbb{Z}_{>0}$, and any $\eta \in (0, 1/2)$, there exists some $d_{\mathrm{IM}} = \Theta(\frac{\log n}{\eta^2})$ such that if we sample a random linear map*

$$\Pi : \mathbb{R}^m \to \mathbb{R}^{d_{\mathrm{IM}}}$$

represented by a $d_{\mathrm{IM}} \times m$ matrix whose entries are independent samples from $\mathcal{N}(0, 1/d_{\mathrm{IM}})$, then for every fixed size-n database $Y \subset \mathbb{R}^m$ and every fixed query vector $u \in \mathbb{R}^m$, with probability at least .99 over the sampling of Π, we have simultaneously for all $y \in Y$ that

$$(1-\eta)\|u-y\|_2^2 \le \|\Pi(u-y)\|_2^2 \le (1+\eta)\|u-y\|_2^2. \tag{2}$$

Moreover, for every fixed realization of Π, there is a Las Vegas preprocessing algorithm that constructs an $\ell_2^{d_{\mathrm{IM}}}$-ANN data structure $\mathcal{D}$ with database ΠY. The preprocessing algorithm always outputs a correct data structure and has expected running time

$$n(1/\eta)^{\mathcal{O}(d_{\mathrm{IM}})} \operatorname{poly}(d_{\mathrm{IM}}, \log n).$$

For every query $u' \in \mathbb{R}^{d_{\mathrm{IM}}}$, the query algorithm is deterministic and outputs a point $p \in Y$ satisfying

$$\|u' - \Pi p\|_2 \le (1+\eta) \min_{y \in Y} \|u' - \Pi y\|_2.$$

The space usage is $n \operatorname{poly}\log n \cdot (1/\eta)^{\mathcal{O}(d_{\mathrm{IM}})}$, and the query time is $d_{\mathrm{IM}} \operatorname{poly}\log n$.

We note that the .99 probability in Theorem 2.2 is taken only over the Gaussian projection Π and concerns the distance-preservation event in Theorem 2.2. In contrast, the randomness used to construct the low-dimensional data structure is Las Vegas: it only affects the preprocessing running time but not the correctness: for every outcome of this randomness, the preprocessing algorithm either continues running or outputs a data structure that successfully answers every query $u' \in \mathbb{R}^{d_{\mathrm{IM}}}$.

We describe our data structure in Algorithm 1.

Algorithm 1 For-All ANN Data Structure for ℓ_2

1: **input:** Input size n, dimension d, and error η. Parameters $t, B, d_{\text{IM}} \in \mathbb{Z}_{>0}$ and $C > 0$ to be specified later.
2: Sample $\mathcal{M}$ as described in Lemma 2.1 with parameter C . Let K denote the number of blocks defined by $\mathcal{M}$.

3: **procedure** PREPROCESS($X \subset \mathbb{R}^d$) ▷ Preprocess a size n database X.
4: **for** $i \in [t]$ **do**
5: Let S_i be a size-B set of indices uniformly sampled with replacement from $[K]$.
6: Define an ordered multi-set $T_i := \bigcup_{k \in S_i} I_k$. ▷ Repeated elements are stored separately.
7: Sample an independent random linear map $\Pi_i : \mathbb{R}^{|T_i|} \to \mathbb{R}^{d_{\text{IM}}}$ as described in Theorem 2.2.
8: Initialize the deterministic-query $\ell_2^{d_{\text{IM}}}$-ANN data structure $\mathcal{D}_i$ from Theorem 2.2 with database

$$\left\{ \Pi_i \left(\frac{1}{\sqrt{B}} (\mathcal{M}x)_{T_i} \right) : x \in X \right\}.$$

▷ $\mathcal{D}_i$ also stores the correspondence between each projected point and its original point $x \in X$.

9: **procedure** QUERY($q \in \mathbb{R}^d$)
10: Compute $\mathcal{M}q$.
11: **for** $i \in [t]$ **do**
12: Query $\mathcal{D}_i$ with $\Pi_i \left(\frac{1}{\sqrt{B}} (\mathcal{M}q)_{T_i} \right)$. Denote the corresponding output point in X by p_i.
13: **return** $p := \underset{p_i \text{ for } i \in [t]}{\arg\min} \|q - p_i\|_2$.

2.2 Correctness of Algorithm 1

In our following analysis of Algorithm 1 (until Corollary 2.9), we assume that we first sample $\mathcal{M}$, and condition on the event that the conclusion of Lemma 2.1 holds for every vector in $\mathbb{R}^d$. This event has probability at least .99. We fix $\mathcal{M}$ for the rest of the argument.

Let Γ denote the space of the randomness used by each loop iteration of the PREPROCESS procedure, with the probability distribution induced by this randomized preprocessing. Each element $\gamma \in \Gamma$ consists of:

- randomness used to sample a size-B ordered multiset S of blocks in Line 5;
- randomness used to sample the Gaussian projection $\Pi_\gamma : \mathbb{R}^{|T|} \to \mathbb{R}^{d_{\text{IM}}}$, where $T = \bigcup_{k \in S} I_k$.

We will use $\mathcal{M}_\gamma$ to denote the normalized linear map defined by the set S sampled using γ, i.e.,

$$\mathcal{M}_\gamma v := \frac{1}{\sqrt{B}} (\mathcal{M}v)_T, \qquad T = \bigcup_{k \in S} I_k.$$

Thus, the complete linear map used by the corresponding invocation of the [IM98] data structure is $\Pi_\gamma \mathcal{M}_\gamma : \mathbb{R}^d \to \mathbb{R}^{d_{\text{IM}}}$.

Definition 2.3. *For each arbitrary $q \in \mathbb{R}^d$, we let x_q^* be a nearest neighbor of q in X. For any $\gamma \in \Gamma$, we say that γ is good for q when all three conditions below hold:*

(a). *(Nearest-neighbor non-inflating.)* $\|\mathcal{M}_\gamma(q - x_q^*)\|_2^2 \le (1+\eta)\|q - x_q^*\|_2^2$.

(b). *(Non-shrinking.) For every* $x \in X$,

$$\|\mathcal{M}_\gamma(q - x)\|_2^2 \ge (1-\eta)\|q - x\|_2^2.$$

(c). *(Distance preservation under the [IM98] projection.) For every* $x \in X$,

$$(1-\eta)\|\mathcal{M}_\gamma(q - x)\|_2^2 \le \|\Pi_\gamma \mathcal{M}_\gamma(q - x)\|_2^2 \le (1+\eta)\|\mathcal{M}_\gamma(q - x)\|_2^2.$$

This definition captures whether a random preprocessing (corresponding to one loop iteration on Line 4) is correct for a fixed query $q \in \mathbb{R}^d$. Formally, we make the following claim:

Claim 2.4. *For any* $i \in [t]$ *and any* $q \in \mathbb{R}^d$, *if* γ_i *is good for* q, *then the output* p_i *on Line 12 satisfies*

$$\|q - p_i\|_2 \le (1 + \Theta(\eta))\|q - x_q^*\|_2.$$

Proof. We can derive:

$$\begin{aligned}
\|q - p_i\|_2^2 &\le (1+\Theta(\eta))\|\mathcal{M}_{\gamma_i}(q - p_i)\|_2^2 && \text{(Definition 2.3 (b))}\\
&\le (1+\Theta(\eta))\|\Pi_{\gamma_i}\mathcal{M}_{\gamma_i}(q - p_i)\|_2^2 && \text{(Lower inequality of Definition 2.3 (c))}\\
&\le (1+\Theta(\eta)) \min_{x \in X} \|\Pi_{\gamma_i}\mathcal{M}_{\gamma_i}(q - x)\|_2^2 && \text{(Correctness of } \mathcal{D}_i)\\
&\le (1+\Theta(\eta))\|\Pi_{\gamma_i}\mathcal{M}_{\gamma_i}(q - x_q^*)\|_2^2 &&\\
&\le (1+\Theta(\eta))\|\mathcal{M}_{\gamma_i}(q - x_q^*)\|_2^2 && \text{(Upper inequality of Definition 2.3 (c))}\\
&\le (1+\Theta(\eta))\|q - x_q^*\|_2^2 && \text{(Definition 2.3 (a))}
\end{aligned}$$

as long as $\eta \in (0, 1/2)$. This implies $\|q - p_i\|_2 \le (1 + \Theta(\eta))\|q - x_q^*\|_2$ as claimed. □

Therefore, to show that for all query q the output p on Line 13 is an approximate nearest neighbor, it suffices to show that for every q, at least one γ_i is good. We first show the point-wise argument that for each fixed query, a sampled randomness $\gamma \in \Gamma$ is good with high probability. Then we will use this point-wise guarantee to prove the for-all guarantee.

Lemma 2.5. *For any* $\eta \in (0, 1/2)$, *there exist appropriately chosen*

$$B = \Theta(\log n/\eta^2), \qquad C = \Theta(B/\eta), \qquad \text{and} \qquad d_{\mathrm{IM}} = \Theta(\log n/\eta^2),$$

such that, for every fixed query $q \in \mathbb{R}^d$, $\mathbb{P}_{\gamma \sim \Gamma}[\gamma \text{ is good for } q] \ge .9$.

Proof. We separately bound the failure probabilities of Definition 2.3(a), Definition 2.3(b), and Definition 2.3(c). We will show that each condition fails with probability at most .01. The claimed result then follows by a union bound.

(a) Non-inflating. For each $k \in [K]$, we write $z_k = z_k(q - x_q^*) = (\mathcal{M}(q - x_q^*))_{I_k}$. We define

$$\mathcal{B} := \Big\{ k \in [K] : \Big| \|z_k\|_2^2 - \|q - x_q^*\|_2^2 \Big| \geq \eta \|q - x_q^*\|_2^2 \Big\}.$$

By Lemma 2.1, we know that $|\mathcal{B}| \leq \frac{K}{\eta C}$. Indeed, $\sum_{k=1}^K \Big| \|z_k\|_2^2 - \|q - x_q^*\|_2^2 \Big| \leq \frac{K}{C} \|q - x_q^*\|_2^2$, so there can be at most $\frac{K}{\eta C}$ terms exceeding $\eta \|q - x_q^*\|_2^2$. Because the set S is sampled with replacement, its B elements are independent and each avoids $\mathcal{B}$ with $1 - \frac{|\mathcal{B}|}{K}$ probability. Therefore,

$$\mathbb{P}[S \cap \mathcal{B} = \varnothing] = \left(1 - \frac{|\mathcal{B}|}{K}\right)^B \geq 1 - \frac{B}{\eta C} \geq .99$$

where the last step holds as long as $C \geq 100B/\eta$. To conclude the argument, we also observe that (a) holds when $S \cap \mathcal{B} = \varnothing$.

(b) Non-shrinking. We will show that as long as $C \geq \frac{2}{\eta}$, for any fixed pair of q, x, with $1 - \exp\big(-\Omega(B\eta^2)\big)$ probability, we have that $\|\mathcal{M}_\gamma(q - x)\|_2^2 \geq (1 - \eta)\|q - x\|_2^2$. Then choosing appropriate $B = \Theta(\log n/\eta^2)$ and a taking a union bound implies that (b) holds with .99 chance.

For each $k \in [K]$, we again write $z_k = z_k(q - x) = (\mathcal{M}(q - x))_{I_k}$. We define

$$\lambda_k := \max\left\{ \frac{\|q - x\|_2^2 - \|z_k\|_2^2}{\|q - x\|_2^2}, 0 \right\}.$$

In other words, λ_k counts the non-negative contribution of an index k, scaled by $1/\|q - x\|_2^2$. We observe that $\lambda_k \in [0, 1]$ and for every possible sampled set S and its corresponding $T = \bigcup_{k \in S} I_k$,

$$\frac{1}{B} \sum_{k \in S} \lambda_k \|q - x\|_2^2 \geq \|q - x\|_2^2 - \frac{1}{B} \|(\mathcal{M}(q - x))_T\|_2^2.$$

Therefore, it suffices to show that

$$\mathbb{P}\left[\frac{1}{B} \sum_{k \in S} \lambda_k \geq \eta \right] \leq \exp\big(-\Omega(B\eta^2)\big).$$

Because $k \in S$ are independent samples, by Hoeffding's bound, we have that, for any $\tau > 0$,

$$\mathbb{P}\left[\frac{1}{B} \sum_{k \in S} \lambda_k \geq \tau + \mathbb{E}\left[\frac{1}{B} \sum_{k \in S} \lambda_k \right] \right] \leq \exp\big(-2B\tau^2\big).$$

If $\mathbb{E}\big[\frac{1}{B} \sum_{k \in S} \lambda_k\big] \leq \eta/2$, then we can conclude the proof by setting $\tau = \eta/2$. And this holds because

$$\begin{aligned}\mathbb{E}\left[\frac{1}{B}\sum_{k\in S}\lambda_k\right] &= \frac{1}{K}\sum_{k\in[K]}\max\left\{\frac{\|q-x\|_2^2-\|z_k\|_2^2}{\|q-x\|_2^2},0\right\}\\ &\le \frac{1}{K}\sum_{k\in[K]}\frac{\left|\|q-x\|_2^2-\|z_k\|_2^2\right|}{\|q-x\|_2^2}\\ &\le \frac{1}{C}\\ &\le \eta/2\end{aligned}$$

where the second to last step follows from Lemma 2.1, and the last step follows from the assumption $C \ge \frac{2}{\eta}$.

(c) Distance preservation. We condition on an arbitrary fixed sample S. Under this conditioning, $\mathcal{M}_\gamma$ is fixed, thus $\{\mathcal{M}_\gamma(q-x) : x \in X\}$ is a fixed. The projection Π_γ is sampled independently of S. Thus, by Theorem 2.2, with probability at least .99 over Π_γ, we have simultaneously for every $x \in X$ that

$$(1-\eta)\|\mathcal{M}_\gamma(q-x)\|_2^2 \le \|\Pi_\gamma\mathcal{M}_\gamma(q-x)\|_2^2 \le (1+\eta)\|\mathcal{M}_\gamma(q-x)\|_2^2.$$

Averaging over the randomness of S gives Definition 2.3 (c) with .99 probability.

□

Given the above point-wise guarantee, we next show a for-all guarantee: if we sample t-many independent randomness $\gamma_1, \cdots, \gamma_t \sim \Gamma$ (corresponding to t loop iterations on Line 4) for an appropriate choice of t, then for all $q \in \mathbb{R}^d$, there always exists $\gamma \in \{\gamma_1, \cdots, \gamma_t\}$ that is good for q. In fact, we will show a stronger argument that a large fraction of the sampled randomness is good.

For this purpose, for any $q \in \mathbb{R}^d$, we define the good set $G_q := \{\gamma \in \Gamma : \gamma \text{ is good for } q\}$. We also define the range family $\mathcal{G} = \{G_q : q \in \mathbb{R}^d\}$. Let μ denote the probability distribution on Γ induced by the PREPROCESS procedure. By Lemma 2.5, for every fixed $q \in \mathbb{R}^d$,

$$\mu(G_q) = \mathbb{P}_{\gamma\sim\Gamma}[\gamma \text{ is good for } q] \ge .9. \tag{3}$$

For independent samples $\gamma_1, \cdots, \gamma_t$ and any $G_q \in \mathcal{G}$, we define the empirical measure of G_q by

$$\widehat{\mu}_t(G_q) := \frac{1}{t}\sum_{i=1}^{t}\mathbb{1}[\gamma_i \in G_q].$$

In other words, $\widehat{\mu}_t(G_q)$ is the fraction of the t sampled loop iterations that are good for q. We want to show that uniformly over G_q, this empirical measure $\hat{\mu}$ is close to the true measure μ. For this purpose, we will apply the following uniform convergence theorem to the range space $(\Gamma, \mathcal{G})$.

Theorem 2.6 ([VC71, BEHW89])**.** *Let $(\Gamma, \mathcal{G})$ be a range space with VC-dimension ν, and let μ be any probability distribution on Γ. Let $\gamma_1, \cdots, \gamma_t$ be independent samples from μ, and define*

$$\widehat{\mu}_t(G) := \frac{1}{t} \sum_{i=1}^{t} \mathbb{1}[\gamma_i \in G].$$

for every $G \in \mathcal{G}$. Then there exists an absolute constant $c > 0$ such that for every $\alpha, \delta \in (0,1)$, if

$$t \geq \frac{c}{\alpha^2} \left(\nu \log \frac{2}{\alpha} + \log \frac{2}{\delta} \right),$$

then, with probability at least $1 - \delta$ over $\gamma_1, \cdots, \gamma_t$, it holds simultaneously for every $G \in \mathcal{G}$ that

$$|\widehat{\mu}_t(G) - \mu(G)| \leq \alpha.$$

This is saying that if we set $\alpha = 0.1$ and sample $\gamma_1, \cdots, \gamma_t$ for t proportional to the VC-dimension of $\mathcal{G}$, then

$$\mathbb{P}_{\gamma_1, \cdots, \gamma_t} \left[\forall q \in \mathbb{R}^d : \frac{1}{t} \big| \{ i : \gamma_i \text{ is good for } q \} \big| \geq .8 \right] \geq 1 - \delta.$$

We now bound the VC-dimension of $\mathcal{G}$.

Lemma 2.7. $\mathrm{VCdim}(\mathcal{G}) = \mathcal{O}(d \log n)$.

Proof. We will show that for any set $A = \{\gamma_1, \cdots, \gamma_\nu\} \subset \Gamma$, its intersection with G_q for any fixed q is fully determined by the signs of $n^2 + 4n\nu$ polynomials of degree at most 2 in d real variables, i.e. coordinates of q. Assuming this is true, then we can apply the following bound on the number of realizable sign vectors:

Theorem 2.8 ([AS88])**.** *Let $\mathcal{P}_1, \cdots, \mathcal{P}_s$ be real polynomials in d real variables, each of degree at most D. If $s \geq d$, then the number of realizable sign vectors*

$$(\mathrm{sgn}(\mathcal{P}_1(z)), \cdots, \mathrm{sgn}(\mathcal{P}_s(z)) \in \{-1, 0, +1\}^s) : z \in \mathbb{R}^d$$

is at most $(\frac{8eDs}{d})^d$.

In our application, $s \leq n^2 + 4n\nu$ and $D = 2$. This implies that

$$\left| \left\{ A \cap G_q : q \in \mathbb{R}^d \right\} \right| \leq \left(\frac{16e(n^2 + 4n\nu)}{d} \right)^d. \tag{4}$$

Finally, if $\mathcal{G}$ shatters A, then this further implies that

$$2^\nu \leq \left(\frac{16e(n^2 + 4n\nu)}{d} \right)^d$$

Solving this shows that the maximum ν, i.e. the VC-dimension of $\mathcal{G}$, is $\mathcal{O}(d \log n)$ as claimed.

To complete the proof, we observe that the intersection of A with G_q for any fixed q is determined by the signs of the following five categories of polynomials:

1. $\left\{\mathcal{P}^1_{x,x'} : x, x' \in X\right\}$, which is defined as $\mathcal{P}^1_{x,x'}(q) := \|q - x\|_2^2 - \|q - x'\|_2^2$. The signs of these polynomials fully determine the ordering from the nearest to the furthest neighbors of q. In particular, they determine the *nearest neighbor* x^*_q. There are at most n^2 of them.
2. $\left\{\mathcal{P}^2_{x,i} : x \in X; i \in [\nu]\right\}$, which is defined as $\mathcal{P}^2_{x,i}(q) := \|\mathcal{M}_{\gamma_i}(q-x)\|_2^2 - (1-\eta)\|q-x\|_2^2$. The signs of these polynomials fully determine whether the *non-shrinking* condition holds. There are $n\nu$ of them.
3. $\left\{\mathcal{P}^3_{x,i} : x \in X; i \in [\nu]\right\}$, which is defined as $\mathcal{P}^3_{x,i}(q) := (1+\eta)\|q-x\|_2^2 - \|\mathcal{M}_{\gamma_i}(q-x)\|_2^2$. Once x^*_q is determined by the signs of the first category of polynomials, the signs of these polynomials fully determine whether the *non-inflating* condition holds. There are $n\nu$ of them.
4. $\left\{\mathcal{P}^4_{x,i} : x \in X; i \in [\nu]\right\}$, which is defined as

   $$\mathcal{P}^4_{x,i}(q) := \|\Pi_{\gamma_i}\mathcal{M}_{\gamma_i}(q-x)\|_2^2 - (1-\eta)\|\mathcal{M}_{\gamma_i}(q-x)\|_2^2.$$

   The signs of these polynomials determine whether the lower inequalities under the [IM98] projection hold. There are $n\nu$ of them.
5. $\left\{\mathcal{P}^5_{x,i} : x \in X; i \in [\nu]\right\}$, which is defined as

   $$\mathcal{P}^5_{x,i}(q) := (1+\eta)\|\mathcal{M}_{\gamma_i}(q-x)\|_2^2 - \|\Pi_{\gamma_i}\mathcal{M}_{\gamma_i}(q-x)\|_2^2.$$

   The signs of these polynomials determine whether the upper inequalities under the [IM98] projection hold. There are $n\nu$ of them.

In summary, the number of distinct intersection $A \cap G_q$ across all $q \in \mathbb{R}^d$ is at most the number of distinct realizable sign vectors

$$\left(\operatorname{sgn}(\mathcal{P}(q)) : \mathcal{P} \in \{\mathcal{P}^1_{x,x'}\} \cup \{\mathcal{P}^2_{x,i}\} \cup \{\mathcal{P}^3_{x,i}\} \cup \{\mathcal{P}^4_{x,i}\} \cup \{\mathcal{P}^5_{x,i}\}\right)$$

across all $q \in \mathbb{R}^d$. There are at most $n^2 + 4n\nu$ relevant polynomials. Also, because $\mathcal{M}_{\gamma_i}$ and $\Pi_{\gamma_i}\mathcal{M}_{\gamma_i}$ are fixed linear maps, the maximum degree of these polynomials in coordinates of q is 2 (from taking the square of norms). This justifies Eq. (4) and concludes the proof. □

Finally, we show the correctness guarantee that with .9 success probability over the sampling of $\mathcal{M}$ and the PREPROCESS procedure, Algorithm 1 deterministically outputs an $(1 + \Theta(\eta))$-approximate nearest neighbor for all queries $q \in \mathbb{R}^d$.

Corollary 2.9. *Given parameters n, d, η, and a size-n database X, there exist appropriately chosen*

$$t = \Theta(d \log n), \quad B = \Theta(\log n/\eta^2), \quad C = \Theta(B/\eta), \quad d_{\mathrm{IM}} = \Theta(\log n/\eta^2)$$

such that with .9 success probability over the sampling of $\mathcal{M}$ and the Preprocess *procedure, we have that*

$$\forall q \in \mathbb{R}^d : \|q - p\|_2 \le (1 + \Theta(\eta))\|q - x_q^*\|_2$$

where $p :=$ Query(q).

Proof. We condition on an arbitrary fixed realization of $\mathcal{M}$ such that the conclusion of Lemma 2.1 holds. We apply Theorem 2.6 by plugging in $\nu := \mathrm{VCdim}(\mathcal{G}) = \mathcal{O}(d \log n)$, with $\alpha = .1$ and $\delta = .01$. Then for an appropriate $t = \Theta(d \log n)$, with probability at least .99 over the independent samples $\gamma_1, \cdots, \gamma_t$, it holds simultaneously for every $G_q \in \mathcal{G}$ that

$$|\widehat{\mu}_t(G_q) - \mu(G_q)| \le .1.$$

By Eq. (3), for every $q \in \mathbb{R}^d$, $\mu(G_q) \ge .9$. Therefore, on the above event, simultaneously for every $q \in \mathbb{R}^d$,

$$\frac{1}{t}|\{i \in [t] : \gamma_i \text{ is good for } q\}| = \widehat{\mu}_t(G_q) \ge .8. \tag{5}$$

In particular, for every $q \in \mathbb{R}^d$, there exists at least one $i \in [t]$ such that γ_i is good for q. As a result, the candidate p_i returned on Line 12 satisfies

$$\|q - p_i\|_2 \le (1 + \Theta(\eta))\|q - x_q^*\|_2.$$

The final output p on Line 13 minimizes the true Euclidean distance to q among all candidates $p_1, \cdots, p_t$. Therefore,

$$\|q - p\|_2 \le \|q - p_i\|_2 \le (1 + \Theta(\eta))\|q - x_q^*\|_2.$$

Finally, by a union bound over two failure events (failed sampling of $\mathcal{M}$ and the failure case of Theorem 2.6), each of probability .01, we have that the probability that the claimed guarantee holds simultaneously for every query is at least .9. The query algorithms of the data structures $\mathcal{D}_1, \cdots, \mathcal{D}_t$ are deterministic, and the remaining steps of Query are deterministic as well. Therefore, conditioned on the successful preprocessing event, the claimed approximation guarantee holds deterministically for every $q \in \mathbb{R}^d$.

□

2.3 Efficient outputting step

In this section, we replace Line 13 in Algorithm 1 with a halving procedure that runs in time only linear in d, up to $\mathrm{poly}(\log n, \eta^{-1})$ factors. We crucially rely on the fact that for every query q, there is not only one good sampled iteration, but actually $.8t$ of them, i.e. Eq. (5).

We will use the same sampled sets $S_1, \cdots, S_t$ to estimate the distances from q to the returned candidates $p_1, \cdots, p_t$. Thus, no additional randomness is needed. During the Preprocess procedure, we additionally store $\mathcal{M}x$ for every $x \in X$.

We let $\varepsilon := \frac{\eta}{100 \log t}$. For every $i \in [t]$, every query $q \in \mathbb{R}^d$, and every $x \in X$, we define

$$\widetilde{d}_i(q, x)^2 := \mathrm{median}_{k \in S_i} \|(\mathcal{M}(q - x))_{I_k}\|_2^2.$$

Here, the median counts the elements of the multiset S_i with their multiplicities.

Definition 2.10. *We say that S_i is valid for q if the following holds:*

$$\forall x \in X : (1-\varepsilon)\|q-x\|_2^2 \le \widetilde{d}_i(q,x)^2 \le (1+\varepsilon)\|q-x\|_2^2. \tag{6}$$

In other words, this definition captures whether a sampled set and its induced median estimates the true Euclidean distances for a fixed query q against all database points. We show that, simultaneously for all query $q \in \mathbb{R}^d$, most of the already sampled sets S_i are valid.

The proof parallels our proof strategy in the previous section: we first show the point-wise guarantee that for each fixed query, a sampled set is valid with high probability; then we bound the VC-dimension of the range family (induced by valid samples) by counting the sign vectors of polynomials (Theorem 2.8) and show the for-all guarantee using the uniform convergence theorem (Theorem 2.6).

Lemma 2.11. *For appropriately chosen*

$$B = \Theta(\log n/\eta^2), \qquad C = \Theta(B/\eta), \qquad t = \Theta(d \log n),$$

conditioned on a fixed map $\mathcal{M}$ satisfying the conclusion of Lemma 2.1, with probability at least .99 over $\gamma_1, \cdots, \gamma_t$, it holds simultaneously for every $q \in \mathbb{R}^d$ that

$$\frac{1}{t}|\{i \in [t] : S_i \text{ is valid for } q\}| \ge .95.$$

Proof. As outlined in the paragraph preceding Lemma 2.11, we start with showing the point-wise guarantee for each q, and then show the for-all guarantee.

Point-wise guarantee. We first fix an arbitrary $q \in \mathbb{R}^d$ and show that one sampled set S is valid for q with probability at least .99.

Fix any $x \in X$. For each $k \in [K]$, write $z_k = z_k(q-x) = (\mathcal{M}(q-x))_{I_k}$, and define

$$\mathcal{B}_x := \left\{k \in [K] : \left|\|z_k\|_2^2 - \|q-x\|_2^2\right| > \varepsilon\|q-x\|_2^2\right\}.$$

By Lemma 2.1, $|\mathcal{B}_x| \le \frac{K}{\varepsilon C}$. Also, since $C\varepsilon = \Theta\left(\frac{B}{\log t}\right)$, we may increase the hidden constants in B and C so that $\frac{1}{C\varepsilon} \le \frac{1}{20}$.

Let $k_1, \cdots, k_B$ denote the independent samples forming S, and let

$$Y_j := \mathbb{1}[k_j \in \mathcal{B}_x], \qquad j \in [B].$$

Then $\mathbb{E}[Y_j] \le 1/20$. If $\widetilde{d}(q,x)^2$ does not satisfy Eq. (6), then at least half of the sampled blocks belong to $\mathcal{B}_x$. Therefore, by Hoeffding's bound,

$$\mathbb{P}\left[\widetilde{d}(q,x)^2 \ne (1\pm\varepsilon)\|q-x\|_2^2\right] \le \mathbb{P}\left[\frac{1}{B}\sum_{j=1}^{B} Y_j \ge \frac{1}{2}\right] \le \exp(-\Omega(B)).$$

Taking a union bound over all $x \in X$, and choosing appropriate $B = \Theta(\log n/\eta^2)$ gives $\mathbb{P}_S[S \text{ is valid for } q] \ge .99$.

For-all guarantee. It remains to show this claim hold simultaneously for every $q \in \mathbb{R}^d$. For every $q \in \mathbb{R}^d$, we define $H_q := \{\gamma \in \Gamma : S_\gamma \text{ is valid for } q\}$ and let $\mathcal{H} := \{H_q : q \in \mathbb{R}^d\}$. The above point-wise argument shows that $\mu(H_q) \geq .99$ for every fixed q, where μ is the probability distribution on induced by the PREPROCESS procedure.

We claim that

$$\mathrm{VCdim}(\mathcal{H}) = \mathcal{O}(d \log n).$$

Indeed, for every $x \in X$ and $k \in [K]$, we can define two polynomials

$$\mathcal{Q}^1_{x,k}(q) := \|(\mathcal{M}(q-x))_{I_k}\|_2^2 - (1-\varepsilon)\|q-x\|_2^2$$

and

$$\mathcal{Q}^2_{x,k}(q) := (1+\varepsilon)\|q-x\|_2^2 - \|(\mathcal{M}(q-x))_{I_k}\|_2^2.$$

For any fixed collection of sampled sets, the signs of these polynomials determine which sampled blocks are accurate for every $x \in X$, and hence determine which sampled sets are valid for q. There are at most $2nK$ such polynomials, each of degree at most 2 in the coordinates of q.

Therefore, by the sign vector bound (Theorem 2.8), the number of different intersections induced by $\mathcal{H}$ is at most $\left(\frac{32enK}{d}\right)^d$. If $\mathcal{H}$ shatters a set of size ν', then

$$2^{\nu'} \leq \left(\frac{32enK}{d}\right)^d.$$

Since $K = \mathcal{O}(d)$ and $n \gg d$, this implies $\nu' = \mathcal{O}(d \log n)$ as claimed.

We now apply the uniform convergence theorem, Theorem 2.6, to $(\Gamma, \mathcal{H})$, with $\alpha = .04$ and $\delta = .01$. For an appropriate $t = \Theta(d \log n)$, with probability at least .99,

$$\forall q \in \mathbb{R}^d : \frac{1}{t}|\{i \in [t] : S_i \text{ is valid for } q\}| \geq \mu(H_q) - .04 \geq .95.$$

□

We now describe the modification of Line 13. Let $P = (p_1, \cdots, p_t)$ be the list of candidates returned on Line 12. We repeatedly reduce the number of candidates by half. When the current list has size M, we arbitrarily partition $[t]$ into M disjoint sets

$$V_1, \cdots, V_M, \qquad |V_a| = \frac{t}{M}.$$

For a candidate p_m for $m \in [M]$, we define

$$\mathrm{score}(p_m) := \mathrm{median}_{i \in V_m} \widetilde{d}_i(q, p_m)^2.$$

We retain $M/2$ candidates with the smallest scores (breaking ties arbitrarily). We repeat this procedure until only one candidate remains, and return this candidate. Equivalently, Line 13 is replaced with the following procedure.

Algorithm 2 Halving procedure replacing Line 13 of Algorithm 1

Let $P \leftarrow (p_1, \cdots, p_t)$.
while $|P| > 1$ **do**
  Let $M \leftarrow |P|$ and write $P = (p_1, \cdots, p_M)$.
  Arbitrarily partition $[t]$ into M sets $V_1, \cdots, V_M$, each of size t/M.
  for $m \in [M]$ **do**
    Set $\text{score}(p_m) \leftarrow \text{median}_{i \in V_m} \widetilde{d}_i(q, p_m)^2$.
  Replace P by the $M/2$ candidates with the smallest scores.
return the unique candidate in P.

We next prove that the halving procedure preserves a constant fraction of approximate nearest neighbors.

Lemma 2.12. *Suppose that, for a fixed query q,*

(a). *at least $.8t$ of the initial candidates $p_1, \cdots, p_t$ satisfy $\|q - p_i\|_2 \le (1 + \Theta(\eta))\|q - x_q^*\|_2$;*

(b). *at least $.95t$ of the sampled sets S_i are valid for q.*

Then the candidate returned by the halving procedure satisfies

$$\|q - p\|_2 \le (1 + \Theta(\eta))\|q - x_q^*\|_2.$$

Proof. We prove the following invariant. At the beginning of any round, if the current list contains M candidates p, then at least $.8M$ of them satisfy $\|q - p\|_2 \le R$ for some radius R. The invariant holds initially with

$$R_0 := (1 + \Theta(\eta))\|q - x_q^*\|_2.$$

In the inductive step, consider a round in which the current list has size M, and assume that the invariant holds with radius R. Each group has size $|V_m| = \frac{t}{M}$. Call the score of a candidate p_m accurate if

$$(1 - \varepsilon)\|q - p_m\|_2^2 \le \text{score}(p_m) \le (1 + \varepsilon)\|q - p_m\|_2^2. \tag{7}$$

If more than half of the indices in V_m correspond to valid sampled sets, then the score of p_m is accurate.

There are at most $.05t$ invalid sampled sets. Since the groups $V_1, \cdots, V_M$ are disjoint, the number of candidate groups containing at least $|V_m|/2$ invalid sampled sets is at most

$$\frac{.05t}{|V_m|/2} = \frac{.05t}{t/(2M)} = .1M.$$

Therefore, at most $.1M$ candidates receive inaccurate scores.

By the inductive hypothesis, at least $.8M$ candidates p satisfy $\|q - p\|_2 \le R$. Hence, at least $.7M$ candidates both satisfy $\|q - p\|_2 \le R$ and receive accurate scores. Each such candidate has score at most $(1 + \varepsilon)R^2$. Since we retain $M/2$ candidates with smallest scores after each round, and $.7M > M/2$, the score of every retained candidate is at most $(1 + \varepsilon)R^2$.

By the definition of accurate (Eq. (7)), every retained candidate satisfies $\|q-p\|_2^2 \le \frac{1+\varepsilon}{1-\varepsilon}R^2$. Among the $M/2$ retained candidates, at most $.1M$ have inaccurate scores. Therefore, at least $.4M$ retained candidates satisfy $\|q-p\|_2^2 \le \frac{1+\varepsilon}{1-\varepsilon}R^2$, which is .8-fraction of the candidates for the next round. This proves the invariant.

After $\log t$ rounds, only one candidate remains. By the invariant, this candidate satisfies

$$\|q-p\|_2 \le \left(\sqrt{\frac{1+\varepsilon}{1-\varepsilon}}\right)^{\log t}(1+\Theta(\eta))\|q-x_q^*\|_2.$$

Since we set $\varepsilon = \eta/(100\log t)$ and $\eta \in (0,1/2)$,

$$\left(\sqrt{\frac{1+\varepsilon}{1-\varepsilon}}\right)^{\log t} \le \exp(\eta/50) = 1+\mathcal{O}(\eta).$$

Therefore, $\|q-p\|_2 \le (1+\Theta(\eta))\|q-x_q^*\|_2$ as claimed.

□

We conclude with the correctness and running time of the modified algorithm. The following corollary immediately gives our main result (Theorem 1.1).

Corollary 2.13. *Replace Line 13 in Algorithm 1 with the halving procedure in Algorithm 2. For appropriately chosen*

$$t = \Theta(d\log n), \qquad B = \Theta(\log n/\eta^2), \qquad C = \Theta(B/\eta), \qquad d_{\mathrm{IM}} = \Theta(\log n/\eta^2),$$

with probability at least .9 over the sampling of $\mathcal{M}$ and the Preprocess *procedure, the resulting data structure satisfies*

$$\forall q \in \mathbb{R}^d : \|q - \textsc{Query}(q)\|_2 \le (1+\Theta(\eta))\|q-x_q^*\|_2.$$

Moreover, the data structure takes $\mathrm{poly}(n,d)$ *space; the query algorithm is deterministic and runs in time*

$$d\,\mathrm{poly}(\log n, \eta^{-1}).$$

Proof. The correctness holds by combining Lemma 2.12 with Eq. (5) (condition (a) of Lemma 2.12) and Lemma 2.11 (condition (b) of Lemma 2.12). The space bound holds since $t = \mathrm{poly}(n,d)$ and each [IM98] data structure takes polynomial space.

It remains to bound the query time. Let

$$l_{\max} := \max_{k\in[K]} l_k = \mathcal{O}(C^3 \,\mathrm{poly}\log d).$$

Computing one value $\widetilde{d}_i(q,x)^2$ requires reading B blocks and takes time

$$\mathcal{O}(Bl_{\max}) = \mathcal{O}(BC^3\,\mathrm{poly}\log d).$$

In each halving round, every $i \in [t]$ is used to validate exactly one candidate. Thus, one round takes $\mathcal{O}(tBC^3 \operatorname{poly}\log d)$ time. There are $\log_2 t = \mathcal{O}(\log n)$ rounds, so the total time of the halving procedure is

$$\mathcal{O}(tBC^3 \operatorname{poly}\log d \log t) = d \operatorname{poly}(\log n, \eta^{-1}).$$

The medians and the set of candidates with the smallest scores can be computed in linear time in the number of values considered, and do not change this bound.

Computing $\mathcal{M}q$ and querying $\mathcal{D}_1, \cdots, \mathcal{D}_t$ also take $d \operatorname{poly}(\log n, \eta^{-1})$ time. Hence the total query time is $d \operatorname{poly}(\log n, \eta^{-1})$.

□

AI Disclosure: We used Claude Opus 4.8 to assist with writing and grammar checks, and GPT 5.6 Sol to audit the proofs, as well as to generate the efficient outputting procedure in Section 2.3. The tools materially affected Section 2.3. The authors verified the correctness and originality of all content including references.

References

[AS88] Noga Alon and Edward R Scheinerman. Degrees of freedom versus dimension for containment orders. *Order*, 5(1):11–16, 1988.

[BEHW89] Anselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Learnability and the vapnik–chervonenkis dimension. *Journal of the ACM*, 36(4):929–965, 1989.

[Cla88] Kenneth L. Clarkson. A randomized algorithm for closest-point queries. *SIAM Journal on Computing*, 17(4):830–847, 1988.

[FI26] Ying Feng and Piotr Indyk. Fast and compact random mappings with uniform guarantees and applications. In *Proceedings of the 58th Annual ACM Symposium on Theory of Computing*, STOC '26, pages 687–697, New York, NY, USA, 2026. Association for Computing Machinery.

[IM98] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 604–613, 1998.

[KOR98] Eyal Kushilevitz, Rafail Ostrovsky, and Yuval Rabani. Efficient search for approximate nearest neighbor in high dimensional spaces. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 614–623, 1998.

[Mei93] Stefan Meiser. Point location in arrangements of hyperplanes. *Information and Computation*, 106(2):286–303, 1993.

[VC71] Vladimir N. Vapnik and Alexey Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and Its Applications*, 16(2):264–280, 1971.